

FRACTUM[®]
AUTOMOTIVE - RF - MICROWAVE

UBEE[®]
Rádio completo no padrão IEEE 802.15.4

Módulo RF U-BEEMAX

V 2.0

Protocolo IEEE 802.15.4

REV 14-04092014

Fractum Indústria e Comércio de Equipamentos Eletrônicos LTDA - Av. Antônio Américo Junqueira 335 Pôr
do Sol - Santa Rita do Sapucaí-MG - CEP 37540-000 - TEL: 55 35 3471 0019

Site: www.fractumrf.com - Email: contato@fractumrf.com

1-MÓDULO U-BEE

O módulo de RF U-Bee foi desenvolvido para trabalhar sob a especificação IEEE 802.15.4 atendendo às necessidades de baixo custo, baixo consumo de energia, e monitoramento de sensores em redes sem fio. O módulo requer mínimo consumo para operação e oferece comunicação e troca de dados entre dispositivos de forma segura. O U-Bee trabalha na banda de frequência ISM 2,4GHz.

1.1-CARACTERÍSTICAS ESPECIAIS

Módulo RF

- Alcance Indoor/Centros urbanos: até 300m (Antena 3dBi Omni)
- Alcance Outdoor Visada Direta: até 1300m (Antena 3dBi Omni)
- Potência de Transmissão: 158mW (+22 dBm)
- Sensibilidade de Recepção: -102 dBm
- Taxa de Dados: 250 Kbps

Rede e Segurança

- Suporta reenvio de mensagens e recebimento de ACKs (Acknowledgements)
- DSSS (Direct Sequence Spread Spectrum)
- Endereçamento de Fonte/Destino
- Comunicações Broadcast e Unicast
- Suporta topologias Ponto-a-Ponto, Ponto-a-Multiponto, Peer-to-Peer, Mesh

Baixo Consumo

- Corrente de Transmissão: 250mA
- Corrente de Recepção: 60mA
- Consumo em Sleep: 50uA

Conversor Analógico-Digital

- 6 Entradas para Conversão Analógica para Digital de 10bits

Facilidade de uso

- Simplicidade de montagem
- Pequeno formato do módulo
- Modo de comandos AT para configuração dos parâmetros do módulo
- Modo API – Application Programming Interface
- Variedade de Comandos

1.2-ESPECIFICAÇÕES

A tabela abaixo mostra as especificações técnicas para o módulo U-Bee:

| Desempenho |
|--|
| Indoor/Centros Urbanos - 300m (Antena 3dBi Omni) |
| Visada Direta - 1300m (Antena 3dBi Omni) |
| Potencia máxima de saída de transmissão – 158mW (+22dBm) |
| Taxa de Dados RF – 250 Kbps |
| Taxa de Dados Serial (Configurável por Software) - 2400-115200 bps |
| Sensibilidade Receptor - -102 dBm |
| Especificações Técnicas |
| Tensão de Operação – 3-3.6V (3.3V típico) |
| Consumo de corrente (típico): |
| <ul style="list-style-type: none">• Modo TX – 250mA• Modo RX – 60mA |
| Especificações Gerais |
| Frequência de Operação – ISM 2.4GHz |
| Dimensões – 48mm x 25mm |
| Faixa de temperatura: -40 °C a +85 °C (Industrial) |
| Especificações de Rede (Networking) |
| Topologias de Rede – Ponto a ponto, Ponto - multiponto, Peer-to-peer, Mesh |
| Canal de Operação Configurável por Software – 16 Canais sequenciais |
| Opções de endereçamento – Rede, Canal e Endereços |

1.3-DESENHO MECÂNICO

A figura abaixo mostra o desenho mecânico para o módulo de RF U-Bee.

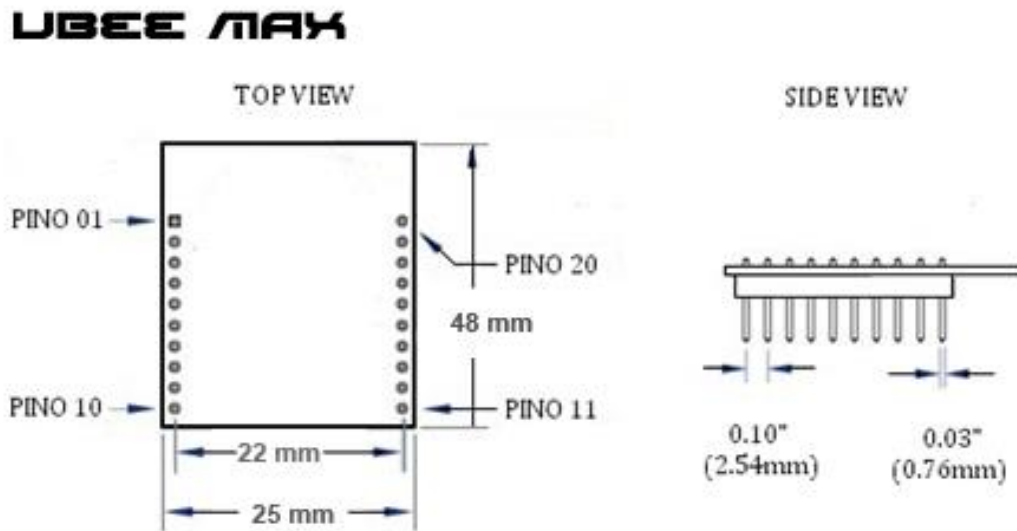


Figura 1.01 – Desenho Mecânico U-Bee

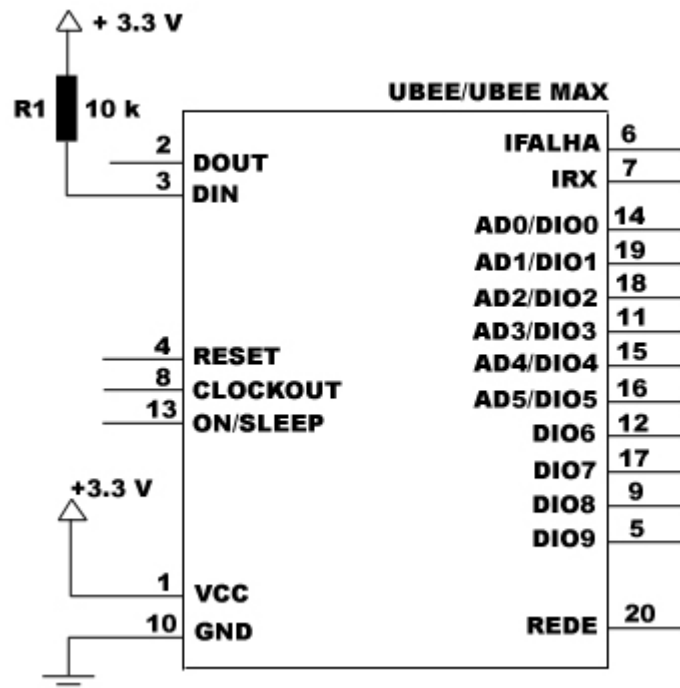
1.4-PINAGEM

A tabela abaixo mostra a pinagem para o módulo U-Bee:

| Pino | Nome | Direção | Descrição |
|------|------------|---------------|--|
| 1 | VCC | - | Alimentação + 3,3V |
| 2 | DOUT | Saída | Saída de dados da USART |
| 3 | DIN | Entrada | Entrada de dados da USART |
| 4 | RESET | Entrada | Inicializa módulo (um pulso nível 0 de pelo menos 200ms) |
| 5 | DIO9 | Entrada/Saída | Entrada/Saída Digital 9 |
| 6 | IFALHA | Saída | Indicação de Falha na Comunicação |
| 7 | IRX | Saída | Indicação de dado recebido |
| 8 | CLOCKOUT | Saída | Saída de Clock de 4MHz |
| 9 | DIO8 | Entrada/Saída | Entrada / Saída Digital 8 |
| 10 | GND | - | Terra |
| 11 | AD3/DIO3 | Entrada/Saída | Entrada Analógica 3 / Entrada / Saída Digital 3 |
| 12 | DIO6 | Entrada/Saída | Entrada/Saída Digital 6 |
| 13 | ON / SLEEP | Entrada | Modo de Operação |
| 14 | AD0/DIO0 | Entrada/Saída | Entrada Analógica 0 / Entrada / Saída Digital 0 |
| 15 | AD4/DIO4 | Entrada/Saída | Entrada Analógica 4 / Entrada / Saída Digital 4 |
| 16 | AD5/DIO5 | Entrada/Saída | Entrada Analógica 5 / Entrada / Saída Digital 5 |
| 17 | DIO7 | Entrada/Saída | Entrada / Saída Digital 7 |
| 18 | AD2/DIO2 | Entrada/Saída | Entrada Analógica 2 / Entrada / Saída Digital 2 |
| 19 | AD1/DIO1 | Entrada/Saída | Entrada Analógica 1 / Entrada / Saída Digital 1 |
| 20 | REDE | Saída | Indicação de REDE |

Montagem:

Conexão mínima: VCC, GND, DIN:



O resistor R1 apenas garante que na ausência de conexões seriais o dispositivo tenha um valor fixo em sua entrada de dados serial. Para configurar o dispositivo não é necessário retirar o resistor de pull-up (R1). Caso já possua um circuito de comunicação serial conectado permanentemente ao módulo U-Bee este resistor poderá ser dispensado. Este resistor pode ser de 10K a 150Kohms.

Conexão para atualização de firmware: VCC, GND, DOUT, DIN.

O módulo inclui um resistor de Pull-up de 150Kohms no pino de **RESET**.

1.5-CARACTERÍSTICAS ELÉTRICAS

Características DC (3.6Vdc):

| Símbolo | Característica | Condição | Mínimo | Típico | Máximo | Unidade |
|-------------|----------------------|----------|--------|--------|--------|---------|
| I_{TX} | Corrente Transmissão | VCC=3.3V | - | 33 | - | mA |
| I_{RX} | Corrente Recepção | VCC=3.3V | - | 28 | - | mA |
| I_{SLEEP} | Corrente em Sleep | VCC=3.3V | 50 | 95 | 200 | uA |

Características Conversor Analógico-Digital (2.4-3.6Vdc):

| Símbolo | Característica | Condição | Mínimo | Típico | Máximo | Unidade |
|---------|---|----------|---------|--------|-----------|---------|
| NR | Resolução | - | - | 10 | - | bits |
| ET | Erro total | - | - | 1 | ±3 | LSb |
| VAIN | Tensão Analógica de entrada | - | VSS-0.3 | - | VCC + 0.3 | V |
| TX | Impedância da fonte de tensão analógica | - | - | - | 10 | kΩ |

2-FUNCIONAMENTO MÓDULO U-BEE

2.1-COMUNICAÇÃO SERIAL

O módulo UBee faz a interface com o dispositivo host através de uma porta serial assíncrona. Através dela, o módulo pode se comunicar com qualquer dispositivo que seja compatível com os níveis lógicos e de tensão da USART; ou através de um conversor de nível para qualquer dispositivo (Exemplo: RS-232, RS485 ou interface USB).

2.2-FLUXO DE DADOS USART

Dispositivos que possuem interface USART podem se conectar diretamente com os pinos do módulo RF assim como mostra a figura abaixo:

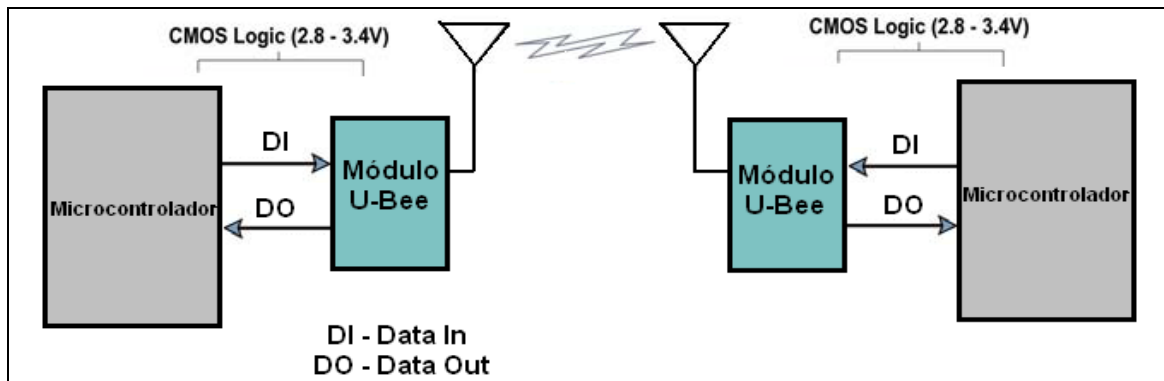
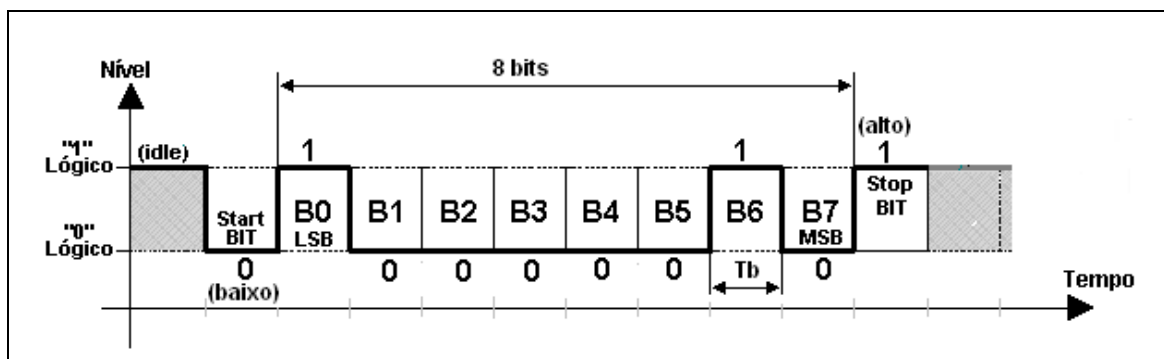


Diagrama do fluxo de dados da interface USART

Os dados entram na USART através do pino DI (pino 03) como um sinal serial assíncrono. Quando não houver dado sendo transmitido (idle), o pino deve permanecer em nível lógico alto. Cada byte de dados consiste em um start bit (baixo), 8 bits de dados (bit menos significativo primeiro) e um stop bit (alto). A figura a seguir ilustra a transmissão do caractere "A":



Transmissão da letra A (0x41 hex)

A USART executa algumas tarefas necessárias para a comunicação, como cronometragem e verificação de paridade. Por isso, a comunicação serial exige que as duas USARTs estejam configuradas igualmente (baud rate, paridade, número de start e stop bits, e o tamanho da palavra).

2.3-MODO TRANSPARENTE

Por padrão, o módulo U-Bee trabalha no Modo Transparente, ou também chamado de Modo Modem. Quando operando neste modo, o módulo atua como uma substituição à linha serial, ou seja, todos os dados recebidos na USART através do pino DI são enfileirados para a transmissão em RF. Quando a informação é recebida por RF, o dado é enviado para o pino DO.

Empacotamento de dados seriais:

Os dados vindos do pino DI são armazenados no buffer DI até que uma das seguintes possibilidades cause a transmissão dos dados via RF:

Nenhum caractere recebido durante um tempo determinado pelo parâmetro $RO*10ms$ (Timeout para empacotamento dos dados). Se $RO=0$, a transmissão ocorre assim que um caractere é recebido.

Se o módulo não puder transmitir os dados imediatamente (por exemplo, se estiver recebendo dados via RF) os dados seriais são armazenados no buffer DI. Quando estiver em condição de transmitir, os dados são empacotados e enviados assim que ocorra um estouro de tempo pelo parâmetro RO , ou assim que o buffer DI atinja seu limite.

Se o buffer DI completar seu limite, o controle de fluxo por hardware ou software deve ser feito para que não haja overflow.

2.4 - DIAGRAMA INTERNO DO FLUXO DE DADOS

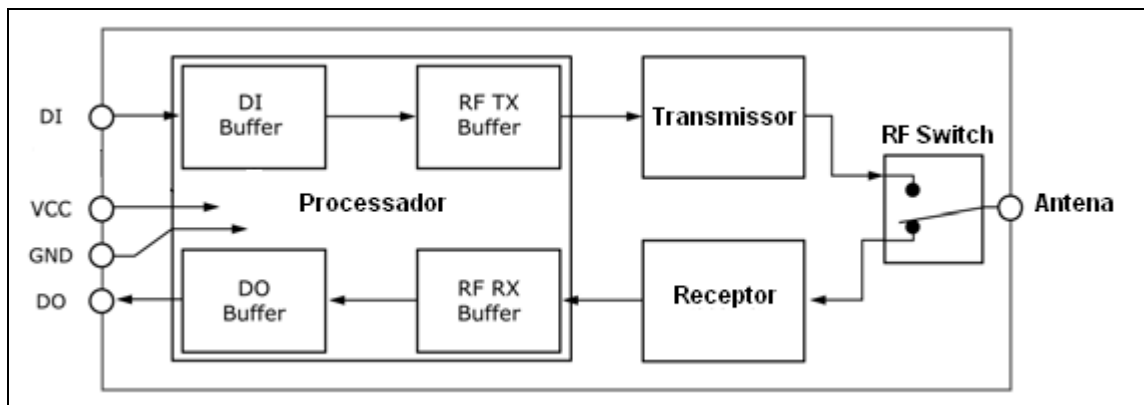


Diagrama interno do fluxo de dados

Data In Buffer (DI Buffer) – Os dados que chegam ao módulo RF através do pino DI (pino 03) são armazenados no DI Buffer até que possam ser processados.

Para que não haja problemas de Overflow, procure enviar mensagens que tenham tamanho menor que o do DI Buffer (100 bytes). Também, faça a interface com uma taxa de comunicação menor (parâmetro BD, Taxa de Dados da Interface) que a taxa de dados de saída.

Um fator que pode causar Overflow é o recebimento de dados via RF continuamente. Os dados seriais vindos do pino DI são armazenados no Buffer DI, porém só poderão ser processados e transmitidos pelo ar quando o módulo não estiver mais recebendo dados da rede.

Data Out Buffer (DO Buffer) – Quando os dados são recebidos via RF, os dados são enviados ao DO Buffer e são enviados ao dispositivo host através do pino de saída DO da porta serial. Uma vez que o DO Buffer atinge sua capacidade máxima, qualquer outro dado recebido via RF é perdido.

O que pode causar Overflow no DO Buffer é quando a taxa de dados via RF é maior que a taxa de dados da interface de saída do módulo, ou seja, o módulo receberá dados vindos do módulo transmissor mais rápido do que ele pode enviar ao dispositivo host.

2.5 – MODO API – APPLICATION PROGRAMMING INTERFACE

O Modo API é um modo de operação opcional em relação ao modo transparente. Através do Modo API é possível ler/configurar todos os parâmetros do módulo e enviar dados seriais através de um protocolo simples e de fácil utilização.

Os frames são divididos como:

- Frames de configuração
- Frames de envio dados
- Frame de recepção de dados
- Frame de resposta ao comando

Este modo de operação é um método simples para enviar dados e realizar configurações no módulo sem a necessidade de entrar no modo de comando.

3-ENDEREÇAMENTO UBEE

Cada pacote de dados enviado contém um campo de Endereço de Origem e Endereço de Destino em seu cabeçalho.

Os módulos UBee, podem assumir a seguinte faixa de endereço:

- Endereço de Fonte (SA) : 0 à 254
- Endereço de Destino (DA): 0 à 255

3.1-MODO UNICAST

O Modo Unicast é a forma que o módulo se comunica somente com outro módulo. Este modo é o único capaz de reenviar uma mensagem com confirmação de recebimento (ACK). Enquanto estiver operando neste modo, os módulos receptores enviam um ACK (reconhecimento) da recepção do pacote de RF ao transmissor. Se o módulo transmissor não receber o ACK, ele reenviará o pacote de dados até o número de vezes configurado em ATNR ou até que o módulo receba o ACK.

Para operar neste modo, o módulo deve estar configurado com um endereço de destino (ATDA) diferente de 255.

3.2-MODO BROADCAST

Este modo de operação é quando se deseja enviar uma mensagem para todos os módulos que estiverem ao alcance da transmissor. Neste modo, não é possível enviar ou receber uma mensagem de confirmação (ACK).

Qualquer módulo dentro do alcance da rede irá aceitar um pacote contendo um endereço Broadcast(ATDA 255). Para enviar um pacote Broadcast para todos os módulos configure o endereço de destino de todos os módulos como 255 (ATDA 255).

3.3-IDENTIFICADOR DE REDE

Cada módulo UBee possui um endereço de fonte (SA), um endereço de destino (DA) e um endereço de rede, ou identificador de rede. Este identificador de rede serve para agrupar vários módulos em somente uma rede única.

O identificador de rede é configurado pelo comando ATID, e pode variar de 0 à 255. Todos os módulos de uma mesma rede, devem ter o mesmo identificador.

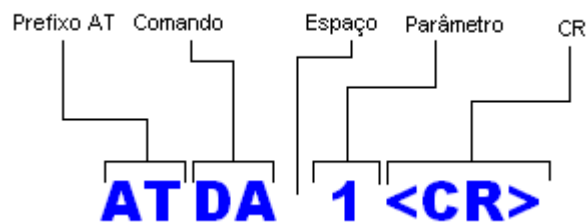
4-MODO DE COMANDO AT

Para ler ou modificar os parâmetros do módulo UBee, o módulo deve primeiro entrar no Modo Comando.

Para entrar no modo de comando AT, deve-se enviar 3 caracteres sequenciais (“+++”) e depois “Enter”. Em ASCII deve ser enviado os seguintes caracteres:

0x2B 0x2B 0x2B 0x0D

Para enviar comandos AT, deve-se enviar a sintaxe como mostra a figura abaixo:



Sintaxe para comandos

Para salvar o novo valor em uma memória não volátil, envie o comando ATWR. Isso fará com que os valores dos parâmetros continuem sendo os parâmetros de configuração do módulo mesmo após uma reinicialização do módulo.

Para sair do Modo de Comando AT, deve-se enviar o comando **ATCN + ENTER**.

Resposta do Sistema

Quando um comando for enviado ao módulo, ele irá analisar e executar o comando. Após executá-lo com sucesso, o módulo retorna uma mensagem “OK”. Se a execução do comando resultar em um erro, o módulo irá retornar uma mensagem de “ERRO”.

5- TABELA DE COMANDOS AT

Os comandos estão contidos dentro das seguintes categorias:

- Especiais
- Rede
- Interface Serial
- Configuração I/O
- Diagnósticos de rede
- Opções de Comandos
- Sleep/Relatório

É aconselhável utilizar a mesma versão de firmware para os módulos operando em uma mesma rede.

Comandos Especiais

| Comando AT | Categoria Comando | Nome e Descrição | Faixa de Valores | Valor Padrão |
|------------|-------------------|---|--------------------------------|--------------|
| WR | Especiais | Write. Escreve o valor do parâmetro em uma memória não volátil fazendo com que as modificações nos parâmetros continuem mesmo após um reset ou Power-up. | - | - |
| RF | Especiais | Restore Fábrica. Restaura os valores de fábrica dos parâmetros. | - | - |
| RE | Especiais | Reinicia o módulo. | - | - |
| RR | Especiais | Reinicia o módulo remoto. Caso DA seja 255 reinicia todos os módulos da rede | - | - |
| PW | Especiais | Power. Atenua a potência de transmissão do rádio com passo de 1dB. | 0-31 | 0 |
| SM | Especiais | Mensagem Inicial. Ativa ou desabilita a mensagem inicial | 0 - Desabilita 1 - Habilita | 1 |
| DF | Especiais | Data de Fabricação. Le a data de fabricação do módulo | mmaaaa | - |

Comandos Rede

| Comando AT | Categoria Comando | Nome e Descrição | Faixa de Valores | Valor Padrão |
|------------|----------------------|--|------------------|--------------|
| CH | Rede (Endereçamento) | Canal de Operação. Lê/Configura o número do canal de operação utilizado para transmitir e receber mensagens entre os módulos de RF (utiliza os números de canais do protocolo IEEE 802.15.4). | 11 – 26 | 11 |

| | | | | |
|------------|----------------------|---|--------------------------------|-----|
| ID | Rede (Endereçamento) | PAN ID. Lê/Configura o PAN ID da rede (Personal Area Network Identification). | 1-255 | 1 |
| DA | Rede (Endereçamento) | Endereço de Destino. Lê/Configura o endereço de destino da mensagem que se deseja. Configurar ATDA 255 ativa o modo broadcast. | 0-255 | 255 |
| SA | Rede (Endereçamento) | Endereço de Fonte. Lê/Configura o endereço de fonte de onde se deseja que parta a mensagem do módulo de RF. | 0-254 | 0 |
| REN | Rede (Roteamento) | Habilita modo roteamento. Todas as mensagens recebidas, serão retransmitidas caso não seja este módulo o destino. | 0 - Desabilita 1 - Habilita | 0 |
| TR | Rede (Configuração) | Tempo de Retransmissão. Configura o tempo de retransmissão de uma mensagem com ACK. O valor configurado é sempre múltiplo de 10ms. | 10 -1000 | 100 |
| NR | Rede (configuração) | Número de Retransmissões. Configura o número de retransmissões de uma mensagem com ACK. | 0 - 20 | 5 |
| AE | Rede (configuração) | ACK Enable. Habilita ou desabilita o envio de ACK (mensagem de confirmação de recebimento da mensagem) | 0-Desabilita 1-Habilita | 1 |

Interface Serial

| Comando AT | Categoria Comando | Nome e Descrição | Faixa de Valores | Valor Padrão |
|------------|-------------------|--|---|--------------|
| BD | Interface Serial | Taxa de Dados da Serial. Lê/Configura o valor da taxa de comunicação entre a porta serial do módulo de RF e o Host. A taxa é alterada somente se for enviado o comando ATWR e reinicializado o módulo. | 0-7 0 = 2400bps 1 = 4800 bps 2 = 9600 bps 3 = 19200 bps 4 = 38400 bps 5 = 57600 bps 6 = 115200 bps 7 = 600bps | 3 |
| RO | Interface Serial | Timeout para empacotamento dos dados. Lê/Configura o tempo de silêncio antes do envio dos dados armazenados no buffer de entrada via RF. Configurando RO = 0 os dados que chegam pela serial são enviados imediatamente. Os dados que estiverem no buffer de entrada serial (para RO>0) são enviados via RF após um time out de RO*10ms quando não receber mais caracteres. | 0-100 | 10 |
| API | Interface Serial | API Enable. Habilita ou desabilita o modo API | 0 – Desabilita 1 – Habilita | 0 |

I/O's

| Comando AT | Categoria Comando | Nome e Descrição | Faixa de Valores | Valor Padrão |
|------------|-------------------|---|------------------|--------------|
| DOI | Configurações I/O | Configura o estado inicial dos I/Os ao ligar o módulo. Se for enviado 0 todos os pinos estarão em nível lógico baixo, se for enviado 1023 todos os pinos estarão em nível lógico alto. | 0-1023 | 0 |
| CT0 | Configurações I/O | Configuração da funcionalidade do terminal. Indica a funcionalidade do terminal: 0 - Saída 1 - Entrada 2 - Saída I/O Line Passing 3 - Entrada I/O Line Passing 4 - Entrada Analógica | 0 - 4 | 4 |
| CT1 | Configurações I/O | Configuração da funcionalidade do terminal. Indica a funcionalidade do terminal: 0 - Saída 1 - Entrada 2 - Saída I/O Line Passing 3 - Entrada I/O Line Passing 4 - Entrada Analógica | 0 - 4 | 4 |
| CT2 | Configurações I/O | Configuração da funcionalidade do terminal. Indica a funcionalidade do terminal: 0 - Saída 1 - Entrada 2 - Saída I/O Line Passing 3 - Entrada I/O Line Passing | 0 - 4 | 4 |

| | | | | |
|-----|-------------------|--|-------------------------------|---|
| | | 4 - Entrada Analógica | | |
| CT3 | Configurações I/O | Configuração da funcionalidade do terminal. Indica a funcionalidade do terminal: 0 - Saída 1 - Entrada 2 - Saída I/O Line Passing 3 - Entrada I/O Line Passing 4 - Entrada Analógica | 0 - 4 | 4 |
| CT4 | Configurações I/O | Configuração da funcionalidade do terminal. Indica a funcionalidade do terminal: 0 - Saída 1 - Entrada 2 - Saída I/O Line Passing 3 - Entrada I/O Line Passing 4 - Entrada Analógica | 0 - 4 | 4 |
| CT5 | Configurações I/O | Configuração da funcionalidade do terminal. Indica a funcionalidade do terminal: 0 - Saída 1 - Entrada 2 - Saída I/O Line Passing 3 - Entrada I/O Line Passing 4 - Entrada Analógica | 0 - 4 | 4 |
| CT6 | Configurações I/O | Configuração da funcionalidade do terminal. Indica a funcionalidade do terminal: 0 - Saída 1 - Entrada 2 - Saída I/O Line Passing 3 - Entrada I/O Line Passing | 0 - 3 | 0 |
| CT7 | Configurações I/O | Configuração da funcionalidade do terminal. Indica a funcionalidade do terminal: 0 - Saída 1 - Entrada 2 - Saída I/O Line Passing 3 - Entrada I/O Line Passing | 0 - 3 | 0 |
| CT8 | Configurações I/O | Configuração da funcionalidade do terminal. Indica a funcionalidade do terminal: 0 - Saída 1 - Entrada 2 - Saída I/O Line Passing 3 - Entrada I/O Line Passing | 0 - 3 | 0 |
| CT9 | Configurações I/O | Configuração da funcionalidade do terminal. Indica a funcionalidade do terminal: 0 - Saída 1 - Entrada 2 - Saída I/O Line Passing 3 - Entrada I/O Line Passing | 0 - 3 | 0 |
| MLP | Configurações I/O | Ativa o envio do I/O Line passing por mudança de estado . Colocando em 0 desativa o envio por mudança de estado, colocando em 1 ativa o envio por mudança de estado no pino. Para habilitar o modo I/O line passing é essencial que habilite o ATMLP 1 | 0 - Desabilita 1- Habilita | 0 |
| PLP | Configurações I/O | Ativa o envio do I/O Line passing por modo periódico Habilita o envio do estado do I/O no modo periódico sendo o valor estipulado sendo múltiplo de 10ms. O Modo é desabilitado colocando 65535. Ex1 : ATPLP 100 Corresponde ao envio de dados periódicos de Line passing a cada 1 segundo Ex2 : ATPLP 65535 Desabilita o envio de I/O Linepassing periódico | 0 65535 | 0 |
| RLI | I/O | Leitura local dos I/Os Faz a leitura de todos os I/O's, Ex1: ATRLI 1111100000 OK Nesse caso todos os I/Os foram configurados como entrada | - | - |

| | | | | |
|------|-----|---|----------|---|
| | | Ex2: ATRLI --00000000 OK Nesse caso, os I/O's 0 e 1 não são entradas. | | |
| RLI0 | I/O | Leitura local do I/O 0 | - | - |
| RLI1 | I/O | Leitura local do I/O 1 | - | - |
| RLI2 | I/O | Leitura local do I/O 2 | - | - |
| RLI3 | I/O | Leitura local do I/O 3 | - | - |
| RLI4 | I/O | Leitura local do I/O 4 | - | - |
| RLI5 | I/O | Leitura local do I/O 5 | - | - |
| RLI6 | I/O | Leitura local do I/O 6 | - | - |
| RLI7 | I/O | Leitura local do I/O 7 | - | - |
| RLI8 | I/O | Leitura local do I/O 8 | - | - |
| RLI9 | I/O | Leitura local do I/O 9 | - | - |
| WLO | I/O | Escreve em todas as saídas locais. Ex.: ATWLO 1023 Escreve 1023 (1023 em binário é: 111111111) em todo o portal DIO. | 0-1023 | - |
| WLO0 | I/O | Escreve no DIO0 Ex.: ATWLO0 1 Coloca em nível lógico alto o DIO0 | 0 - 1 | - |
| WLO1 | I/O | Escreve no DIO1 Ex.: ATWLO1 1 Coloca em nível lógico alto o DIO1 | 0 - 1 | - |
| WLO2 | I/O | Escreve no DIO2 Ex.: ATWLO2 1 Coloca em nível lógico alto o DIO2 | 0 - 1 | - |
| WLO3 | I/O | Escreve no DIO3 Ex.: ATWLO3 1 Coloca em nível lógico alto o DIO3 | 0 - 1 | - |
| WLO4 | I/O | Escreve no DIO4 Ex.: ATWLO4 1 Coloca em nível lógico alto o DIO4 | 0 - 1 | - |
| WLO5 | I/O | Escreve no DIO5 Ex.: ATWLO5 1 Coloca em nível lógico alto o DIO5 | 0 - 1 | - |
| WLO6 | I/O | Escreve no DIO6 Ex.: ATWLO6 1 Coloca em nível lógico alto o DIO6 | 0 - 1 | - |
| WLO7 | I/O | Escreve no DIO7 Ex.: ATWLO7 1 Coloca em nível lógico alto o DIO7 | 0 - 1 | - |
| WLO8 | I/O | Escreve no DIO8 Ex.: ATWLO8 1 Coloca em nível lógico alto o DIO8 | 0 - 1 | - |
| WLO9 | I/O | Escreve no DIO9 Ex.: ATWLO9 1 Coloca em nível lógico alto o DIO9 | 0 - 1 | - |
| RLA | I/O | Lê todas as entradas analógicas locais Faz a leitura de todas as entradas AD's com o valor convertido em decimal: ATRLA ADC0: 612 ADC1: 1023 ADC2: 0 ADC3: 10 ADC4: 55 ADC5: 20 OK | 0 - 1023 | - |

| | | | | |
|-------------|-----|--|----------|---|
| | | | | |
| RLA0 | I/O | Lê a entrada analógica local AD0 Ex: ATRLA0 Faz a leitura do AD0 caso seja uma entrada analógica o módulo retorna o valor convertido em decimal: ATRLA0 611 OK | 0 - 1023 | - |
| RLA1 | I/O | Lê a entrada analógica local AD1 Ex: ATRLA1 Faz a leitura do AD1 caso seja uma entrada analógica o módulo retorna o valor convertido em decimal: ATRLA1 10 OK | 0 - 1023 | - |
| RLA2 | I/O | Lê a entrada analógica local AD2 Ex: ATRLA2 Faz a leitura do AD2 caso seja uma entrada analógica o módulo retorna o valor convertido em decimal: ATRLA2 0 OK | 0 - 1023 | - |
| RLA3 | I/O | Lê a entrada analógica local AD3 Ex: ATRLA3 Faz a leitura do AD3 caso seja uma entrada analógica o módulo retorna o valor convertido em decimal: ATRLA3 525 OK | 0 - 1023 | - |
| RLA4 | I/O | Lê a entrada analógica local AD4 Ex: ATRLA4 Faz a leitura do AD4 caso seja uma entrada analógica o módulo retorna o valor convertido em decimal: ATRLA4 611 OK | 0 - 1023 | - |
| RLA5 | I/O | Lê a entrada analógica local AD5 Ex: ATRLA5 Faz a leitura do AD5 caso seja uma entrada analógica o módulo retorna o valor convertido em decimal: ATRLA5 1023 OK | 0 - 1023 | - |
| RRI | I/O | Lê todas as entradas digitais remotas Lê todas as entradas do módulo remoto cujo endereço (SA) seja igual ao DA deste módulo. Ex: ATRRI Faz a leitura de todos os I/O's: ATRRI NR:0 1111100000 OK Obs.: caso DA esteja configurado como 255(Broadcast) não será executado nenhuma leitura. | 0 - 1 | - |
| RRI0 | I/O | Lê o DIO0 remoto Ex: ATRRI0 Faz a leitura do I/O 0, caso seja uma entrada o módulo retorna o nível lógico dela: ATRRI0 NR:0 1 OK Caso o I/O 0 não seja entrada a resposta será: ATRRI1 NR:0 - OK | 0 - 1 | - |
| RRI1 | I/O | Lê o DIO1 remoto | 0 - 1 | - |
| RRI2 | I/O | Lê o DIO2 remoto | 0 - 1 | - |
| RRI3 | I/O | Lê o DIO3 remoto | 0 - 1 | - |

| | | | | |
|------|-----|--|----------|---|
| RR14 | I/O | Lê o DIO4 remoto | 0 – 1 | - |
| RR15 | I/O | Lê o DIO5 remoto | 0 – 1 | - |
| RR16 | I/O | Lê o DIO6 remoto | 0 – 1 | - |
| RR17 | I/O | Lê o DIO7 remoto | 0 – 1 | - |
| RR18 | I/O | Lê o DIO8 remoto | 0 – 1 | - |
| RR19 | I/O | Lê o DIO9 remoto | 0 – 1 | - |
| RRA | I/O | <p>Lê todas as entradas analógicas remotas Lê todas as entradas analógicas do módulo remoto cujo endereço (SA) seja igual ao DA deste módulo. Ex: ATRRA Faz a leitura de todas as entradas AD's: ATRRA NR:0 ADC0: 413 ADC1: 1023 ADC2: 0 ADC3: 100 ADC4: 550 ADC5: 200 OK</p> | 0 - 1023 | - |
| RRA0 | I/O | <p>Lê a entrada analógica AD0 remota Ex: ATRRA0 Faz a leitura do AD0 caso seja uma entrada analógica o módulo retorna o valor convertido em decimal: ATRRA0 NR:0 413 OK Caso o AD0 não seja entrada analógica, a resposta será: ATRRA0 NR:0 – OK</p> | 0 - 1023 | - |
| RRA1 | I/O | Lê a entrada analógica AD1 remota | 0 - 1023 | - |
| RRA2 | I/O | Lê a entrada analógica AD2 remota | 0 - 1023 | - |
| RRA3 | I/O | Lê a entrada analógica AD3 remota | 0 - 1023 | - |
| RRA4 | I/O | Lê a entrada analógica AD4 remota | 0 - 1023 | - |
| RRA5 | I/O | Lê a entrada analógica AD5 remota | 0 - 1023 | - |
| WRO | I/O | <p>Escreve em todas as saídas remotas. Ex.: ATWLO 1023 Escreve 1023 (1023 em binário é: 111111111) em todo o portal DIO</p> | 0 - 1023 | - |
| WRO0 | I/O | <p>Escreva na saída DIO0 remota EX.: ATWRO0 1 NR:0 OK Coloca em nível lógico alto o DIO0 remoto</p> | 0 – 1 | - |
| WRO1 | I/O | Escreva na saída DIO1 remota | 0 – 1 | - |
| WRO2 | I/O | Escreva na saída DIO2 remota | 0 – 1 | - |
| WRO3 | I/O | Escreva na saída DIO3 remota | 0 – 1 | - |
| WRO4 | I/O | Escreva na saída DIO4 remota | 0 – 1 | - |
| WRO5 | I/O | Escreva na saída DIO5 remota | 0 – 1 | - |
| WRO6 | I/O | Escreva na saída DIO6 remota | 0 – 1 | - |
| WRO7 | I/O | Escreva na saída DIO7 remota | 0 – 1 | - |
| WRO8 | I/O | Escreva na saída DIO8 remota | 0 – 1 | - |
| WRO9 | I/O | Escreva na saída DIO9 remota | 0 – 1 | - |

Diagnósticos

| Comando AT | Categoria Comando | Nome e Descrição | Faixa de Valores | Valor Padrão |
|------------|-------------------|--|--------------------------------|--------------|
| NS | Diagnósticos | Nível do Sinal Recebido. Lê o nível de sinal (em %) dos últimos 4 pacotes recebido (RSSI). O valor mostrado será a média desses pacotes. | 0 – 100% | - |
| IND | Diagnósticos | Indicação de Rede – Sinal Recebido – Habilita a indicação de rede por nível de sinal recebido, pode-se ligar um LED a este pino para visualização de status de rede. Status Sinal Forte – Sinal Médio – Sinal Fraco – Sem Sinal – Forma de visualização LED ligado LED piscando de forma rápida 300 ms LED piscando de forma lenta 800 ms LED Apagado | 0 - Desabilita 1 - Habilita | 1 |
| BAT | Diagnósticos | Nível de tensão de alimentação. Através deste comando pode-se visualizar o nível de tensão da fonte de alimentação do módulo ou bateria. | - | - |
| MSE | Diagnósticos | Mensagem de Status do ACK. Habilita a mensagem de Status de uma transmissão com ACK. | 0 - Desabilita 1 - Habilita | 1 |
| TN | Diagnósticos | Tempo de Notificação. Configura o tempo de envio de pacote de notificação . Este pacote serve para atualizar as indicações de rede. É necessário que todos os módulos tenham o mesmo valor configurado. O parâmetro configurado será multiplicado por 10ms. Este pacote é somente enviado caso não haja nenhum fluxo de dados na rede. | 50 - 1000 | 100 |

Opções de Comandos

| Comando AT | Categoria Comando | Nome e Descrição | Faixa de Valores | Valor Padrão |
|------------|--------------------|---|------------------|--------------|
| CN | Opções Comandos AT | Termina Modo de Comando. Sai do modo de Comando. | - | - |

Sleep/Relatório

| Comando AT | Categoria Comando | Nome e Descrição | Faixa de Valores | Valor Padrão | | | | | | | | | | | | | | | | | | | | | | | | |
|------------|------------------------|--|--|--------------|------|------|------|------|------|------|------|------|------|------|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|--|--|
| MS | Sleep | Modo Sleep. Este comando seleciona se o módulo estará habilitado para operar no modo sleep e de que forma. | 0 = desabilitado 1 = habilitado 2 = habilitado e periódico | 2 | | | | | | | | | | | | | | | | | | | | | | | | |
| TS | Sleep | Tempo de Sleep. Quando selecionado MS = 2, é habilitado o envio do relatório no modo sleep periódico, sendo o valor múltiplo de 8,45s. | 0 - 65535 | 1 | | | | | | | | | | | | | | | | | | | | | | | | |
| PR | Relatório | Ativa o envio do relatório por modo periódico Habilita o envio do relatório para o destino por modo periódico sendo o valor múltiplo de 10ms. O Modo é desabilitado colocando 65535. Ex1 : ATPR 100 Corresponde ao envio de relatório periódico de a cada 1 segundo Ex2 : ATPR 65535 Desabilita o envio do relatório periódico | 1-65535 | 65535 | | | | | | | | | | | | | | | | | | | | | | | | |
| MPR | Relatório local | Ativa o relatório local por modo periódico Habilita o envio do relatório local por modo periódico sendo o valor múltiplo de 10ms. O Modo é desabilitado colocando 65535. Ex1 : ATMPR 100 Corresponde ao envio de relatório periódico de a cada 1 segundo Ex2 : ATMPR 65535 Desabilita o relatório periódico | 1-65535 | 65535 | | | | | | | | | | | | | | | | | | | | | | | | |
| SPR | Relatório Simplificado | Ativa o modo de relatório simplificado. 0 – modo completo 1 – modo simplificado Habilita a impressão do relatório simplificado no seguinte padrão: | 0 - Desabilita 1 - Habilita | 0 | | | | | | | | | | | | | | | | | | | | | | | | |
| | | <table border="1"> <thead> <tr> <th>S</th> <th>Endereço</th> <th>DIO0</th> <th>DIO1</th> <th>DIO2</th> <th>DIO3</th> <th>DIO4</th> <th>DIO5</th> <th>DIO6</th> <th>DIO7</th> <th>DIO8</th> <th>DIO9</th> </tr> </thead> <tbody> <tr> <td>S</td> <td>ADD</td> <td>0-1</td> <td>0-1</td> <td>0-1</td> <td>0-1</td> <td>0-1</td> <td>0-1</td> <td>0-1</td> <td>0-1</td> <td>0-1</td> <td>0-1</td> </tr> </tbody> </table> | S | Endereço | DIO0 | DIO1 | DIO2 | DIO3 | DIO4 | DIO5 | DIO6 | DIO7 | DIO8 | DIO9 | S | ADD | 0-1 | 0-1 | 0-1 | 0-1 | 0-1 | 0-1 | 0-1 | 0-1 | 0-1 | 0-1 | | |
| S | Endereço | DIO0 | DIO1 | DIO2 | DIO3 | DIO4 | DIO5 | DIO6 | DIO7 | DIO8 | DIO9 | | | | | | | | | | | | | | | | | |
| S | ADD | 0-1 | 0-1 | 0-1 | 0-1 | 0-1 | 0-1 | 0-1 | 0-1 | 0-1 | 0-1 | | | | | | | | | | | | | | | | | |

6- MODO DE OPERAÇÃO API – APPLICATION PROGRAMMING INTERFACE

No modo de operação API toda a comunicação USART como módulo UBee se dá através de um protocolo estruturado.

É possível neste modo configurar todos os parâmetros do módulo UBee e ainda enviar dados de forma estruturada e controlada através de um protocolo.

Para habilitar este modo é necessário entrar no modo de comando e enviar ATAPI 1. Por padrão de fábrica o módulo UBee opera no modo transparente, ou seja, todos os dados que estiverem em RX serão enviados ao TX do módulo de destino.

Quando o módulo estiver operando no modo API, todo dado recebido pela serial será tratado como um frame API. Todos os dados do frame API devem ser caracteres em ASCII.

Existem os seguintes tipos de frames API:

Frame de configuração

| | | | | | |
|-------|------|---------|-----------|----------|------|
| START | TIPO | COMANDO | PARÂMETRO | CHECKSUM | STOP |
|-------|------|---------|-----------|----------|------|

Frame de dados com configuração

| | | | | | | | | | |
|-------|------|----|----|----|----|-----|---------|----------|------|
| START | TIPO | SA | DA | ID | CH | ACK | PAYLOAD | CHECKSUM | STOP |
|-------|------|----|----|----|----|-----|---------|----------|------|

Frame de dados sem configuração

| | | | | |
|-------|------|---------|----------|------|
| START | TIPO | PAYLOAD | CHECKSUM | STOP |
|-------|------|---------|----------|------|

Todos os frames possuem os seguintes bytes iguais:

START – 1 byte fixo com valor 0x2A, em ASCII é correspondente ao caractere *

CHECKSUM – 3 bytes, ele é a soma de todos os bytes do pacote de dados menos os bytes de checksum e de stop.

STOP – 1 byte fixo com valor 0x23, em ASCII é correspondente ao caractere #

6.1-FRAME DE CONFIGURAÇÃO

Todo frame API de configuração é formado pelos seguintes campos:

| | | | | | |
|-------|------|---------|-----------|----------|------|
| START | TIPO | COMANDO | PARÂMETRO | CHECKSUM | STOP |
|-------|------|---------|-----------|----------|------|

START – 1 byte fixo com valor 0x2A, em ASCII é correspondente ao caractere *

TIPO – 2 bytes variáveis que determinam o tipo de comando de configuração

PARÂMETRO – 1 à 6 bytes

A estrutura do protocolo é definida com os seguintes campos:

| | | | | | |
|-------|------|---------|-----------|----------|------|
| START | TIPO | COMANDO | PARÂMETRO | CHECKSUM | STOP |
|-------|------|---------|-----------|----------|------|

O START é correspondente ao caractere * em ASCII ou em valor absoluto é 0x2A.

O TIPO define o tipo de comando a ser enviado, podendo ser:

- 01 – Comandos Especias
- 02 - Comandos de REDE
- 03 – Interface Serial
- 04 – I/O's
- 05 – Diagnósticos
- 06 – Sleep/Relatório
- 10 – Mensagem de dados com configuração
- 11 – Mensagem de dados sem configuração
- 00 – Mensagem de Sucesso
- 88 – Mensagem de Erro
- 99 – Mensagem de Erro de Checksum

O COMANDO é similar ao comando AT que se deseja configurar ou ler, por exemplo:
O TIPO 01 pode ter os seguintes comandos WR, RF, RE, RR, PW ou SM.

O PARÂMETRO é o valor que se deseja configurar ou ler.

O CHECKSUM é utilizado para garantir a integridade dos dados. Ele é a soma de todos os bytes do pacote de dados menos os bytes de checksum e de stop.

O STOP é correspondente ao caractere # em ASCII ou em valor absoluto é 0x23.

Exemplos de utilização:

a) Deseja-se configurar o canal 11 como sendo o de operação, portanto, basta enviar:

***02CH11377# (em ASCII)**

| | Start | Tipo | Comando | Parâmetro | Checksum | Stop |
|-----------------------|-------|--------------|--------------|--------------|----------------------|------|
| ASCII | * | 02 | CH | 11 | 377 | # |
| Valor Absoluto | 0x2A | 0x30 0x32 | 0x43 0x48 | 0x31 0x31 | 0x33 0x37 0x37 | 0x23 |

Calculando o CheckSum: $0x2A+0x30+0x32+0x43+0x48+0x31+0x31 = 377$ em decimal

b) Deseja-se configurar o endereço de destino como 001:

***02DA001418#**

| | Start | Tipo | Comando | Parâmetro | Checksum | Stop |
|-----------------------|-------|--------------|--------------|----------------------|----------------------|------|
| ASCII | * | 02 | DA | 001 | 418 | # |
| Valor Absoluto | 0x2A | 0x30 0x32 | 0x44 0x41 | 0x30 0x30 0x31 | 0x34 0x31 0x38 | 0x23 |

Calculando o CheckSum: $0x2A+0x30+0x32+0x44+0x41+0x30+0x30+0x31 = 418$ em decimal

6.1.1-FRAME DE CONFIGURAÇÃO TIPO 01

O frame de configuração tipo 01 corresponde aos comandos AT especiais, e são eles:

| START | Tipo | Comando API | Parâmetro | CHECKSUM | STOP | EXEMPLO | Leitura/Escrita |
|-------|------|-------------|-----------|----------|------|-------------|-----------------|
| * | 01 | WR | 0 | 356 | # | *01WR0356# | Escrita |
| * | 01 | RF | 0 | 339 | # | *01RF0339# | Escrita |
| * | 01 | RE | 0 | 338 | # | *01RE0338# | Escrita |
| * | 01 | RR | 0 | 351 | # | *01RR0351# | Escrita |
| * | 01 | PW | 00 à 31 | xxx | # | *01PW00402# | Leitura/Escrita |
| * | 01 | SM | 0 ou 1 | xxx | # | *01SM0347# | Leitura/Escrita |

Todo frame API enviado ao módulo resulta em uma resposta, podendo ser do tipo:

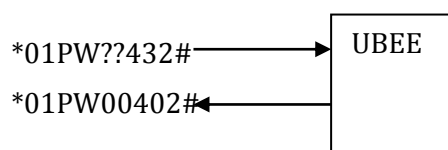
- 00 – Frame de Sucesso
- 01 – Frame de Leitura
- 88 – Frame de Erro
- 99 – Frame de Erro de Checksum

Para realizar a leitura de um certo parâmetro basta enviar no lugar do parâmetro o caractere ? (**0x3F**). Exemplo de uso:

Deseja-se ler o valor configurado em PW, portanto basta enviar *01PW??432#. Observe que são dois caracteres ?, pois o parâmetro deste comando possui no máximo dois caracteres. Ao enviar este frame o módulo responde: *01PW00402#, onde podemos observar os seguintes campos:

| Start | Tipo | Comando | Parâmetro | CheckSum | Stop |
|-------|------|---------|------------|----------|------|
| * | 01 | PW | 00 | 402 | # |
| | | | Valor Lido | | |

Portanto, o valor configurado em PW é 00.



6.1.2-FRAME DE CONFIGURAÇÃO TIPO 02

O frame de configuração tipo 02 corresponde aos comandos AT de rede, e são eles:

| START | Tipo | Comando API | Parâmetro | CHECKSUM | STOP | EXEMPLO | Leitura/Escrita |
|-------|------|-------------|-------------|----------|------|--------------|-----------------|
| * | 02 | CH | 11 à 26 | xxx | # | *02CH11377# | Leitura/Escrita |
| * | 02 | ID | 001 à 255 | xxx | # | *02ID001426# | Leitura/Escrita |
| * | 02 | DA | 000 à 255 | xxx | # | *02DA001418# | Leitura/Escrita |
| * | 02 | SA | 000 à 254 | xxx | # | *02SA001433# | Leitura/Escrita |
| * | 02 | REN | 0 ou 1 | xxx | # | *02REN1418# | Leitura/Escrita |
| * | 02 | TR | 0010 à 1000 | xxx | # | *02TR???558# | Leitura/Escrita |

Todo frame API enviado ao módulo resulta em uma resposta, podendo ser:

- 00 – Frame de Sucesso
- 02 – Frame de Leitura
- 88 – Frame de Erro
- 99 – Frame de Erro de Checksum

6.1.3-FRAME DE CONFIGURAÇÃO TIPO 03

O frame de configuração tipo 03 corresponde aos comandos AT Serias, e são eles:

| START | Tipo | Comando API | Parâmetro | CHECKSUM | STOP | EXEMPLO | Leitura/Escrita |
|-------|------|-------------|-----------|----------|------|--------------|-----------------|
| * | 03 | BD | 0 à 6 | xxx | # | *03BD3326# | Leitura/Escrita |
| * | 03 | RO | 000 à 100 | xxx | # | *03RO010447# | Leitura/Escrita |
| * | 03 | API | 0 à 1 | xxx | # | *03API?422# | Leitura/Escrita |

Todo frame API enviado ao módulo resulta em uma resposta, podendo ser:

- 00 – Frame de Sucesso
- 03 – Frame de Leitura
- 88 – Frame de Erro
- 99 – Frame de Erro de Checksum

6.1.4-FRAME DE CONFIGURAÇÃO TIPO 04

O frame de configuração tipo 04 corresponde aos comandos AT I/O's, e são eles:

| START | Tipo | Comando API | Parâmetro | CHECKSUM | STOP | EXEMPLO |
|-------|------|-------------|---------------|----------|------|-----------------|
| * | 04 | DOI | 0000 à 1023 | xxx | # | *04DOI0000554# |
| * | 04 | CT0 | 0 à 4 | xxx | # | *04CT00389# |
| * | 04 | CT1 | 0 à 4 | xxx | # | *04CT14394# |
| * | 04 | CT2 | 0 à 4 | xxx | # | *04CT24395# |
| * | 04 | CT3 | 0 à 4 | xxx | # | *04CT34396# |
| * | 04 | CT4 | 0 à 4 | xxx | # | *04CT44397# |
| * | 04 | CT5 | 0 à 4 | xxx | # | *04CT54398# |
| * | 04 | CT6 | 0 à 3 | xxx | # | *04CT63398# |
| * | 04 | CT7 | 0 à 3 | xxx | # | *04CT73399# |
| * | 04 | CT8 | 0 à 3 | xxx | # | *04CT80397# |
| * | 04 | CT9 | 0 à 3 | xxx | # | *04CT90398# |
| * | 04 | MLP | 0 ou 1 | xxx | # | *04MLP1424# |
| * | 04 | PLP | 00000 à 65535 | xxx | # | *04PLP01000619# |
| * | 04 | RLI | 0 à 9 | xxx | # | *04RLI1422# |
| * | 04 | WLO0 | 0 ou 1 | xxx | # | *04WLO01481# |
| * | 04 | WLO1 | 0 ou 1 | xxx | # | *04WLO11482# |
| * | 04 | WLO2 | 0 ou 1 | xxx | # | *04WLO21483# |
| * | 04 | WLO3 | 0 ou 1 | xxx | # | *04WLO31484# |
| * | 04 | WLO4 | 0 ou 1 | xxx | # | *04WLO40484# |
| * | 04 | WLO5 | 0 ou 1 | xxx | # | *04WLO50485# |
| * | 04 | WLO6 | 0 ou 1 | xxx | # | *04WLO60486# |
| * | 04 | WLO7 | 0 ou 1 | xxx | # | *04WLO70487# |
| * | 04 | WLO8 | 0 ou 1 | xxx | # | *04WLO80488# |
| * | 04 | WLO9 | 0 ou 1 | xxx | # | *04WLO90489# |
| * | 04 | RLA | 0 à 5 | xxx | # | *04RLA0413# |
| * | 04 | RRI | 0 - 9 | xxx | # | *04RRI9436# |
| * | 04 | RRA | 0 - 5 | xxx | # | *04RRA0419# |
| * | 04 | WRO0 | 0 - 1 | xxx | # | *04WRO00486# |
| * | 04 | WRO1 | 0 - 1 | xxx | # | *04WRO10487# |
| * | 04 | WRO2 | 0 - 1 | xxx | # | *04WRO20488# |
| * | 04 | WRO3 | 0 - 1 | xxx | # | *04WRO30489# |
| * | 04 | WRO4 | 0 - 1 | xxx | # | *04WRO40490# |
| * | 04 | WRO5 | 0 - 1 | xxx | # | *04WRO50491# |
| * | 04 | WRO6 | 0 - 1 | xxx | # | *04WRO60492# |
| * | 04 | WRO7 | 0 - 1 | xxx | # | *04WRO70493# |

| | | | | | | |
|---|----|------|-------|-----|---|--------------|
| * | 04 | WRO8 | 0 - 1 | xxx | # | *04WRO80494# |
| * | 04 | WRO9 | 0 - 1 | xxx | # | *04WRO90495# |

Todo frame API enviado ao módulo resulta em uma resposta, podendo ser:

- 00 – Frame de Sucesso
- 04 – Frame de Leitura
- 88 – Frame de Erro
- 99 – Frame de Erro de Checksum

6.1.5-FRAME DE CONFIGURAÇÃO TIPO 05

O frame de configuração tipo 05 corresponde aos comandos AT de Diagnósticos e são eles:

| START | Tipo | Comando API | Parâmetro | CHECKSUM | STOP | EXEMPLO |
|-------|------|-------------|-------------|----------|------|---------------|
| * | 05 | NS | ? | 352 | # | *05NS?367# |
| * | 05 | IND | 0 ou 1 | xxx | # | *05IND0410# |
| * | 05 | BAT | ? | 421 | # | *05BAT?421# |
| * | 05 | MSE | 0 ou 1 | xxx | # | *05MSE1421# |
| * | 05 | TN | 0050 à 1000 | xxx | # | *05TN0100498# |
| * | 05 | FV | ???? | xxx | # | *05FV????551# |
| * | 05 | HV | ?? | xxx | # | *05HV??427# |

Todo frame API enviado ao módulo resulta em uma resposta, podendo ser:

- 00 – Frame de Sucesso
- 05 – Frame de Leitura
- 88 – Frame de Erro
- 99 – Frame de Erro de Checksum

6.1.6-FRAME DE CONFIGURAÇÃO TIPO 06

O frame de configuração tipo 06 corresponde aos comandos AT de Sleep e Relatório e são eles:

| START | Tipo | Comando API | Parâmetro | CHECKSUM | STOP | Exemplo |
|-------|------|-------------|---------------|----------|------|-----------------|
| * | 06 | MS | 0 à 2 | xxx | # | *06MS0352# |
| * | 06 | TS | 00000 à 65535 | xxx | # | *06TS00010552# |
| * | 06 | PR | 00001 à 65535 | xxx | # | *06PR00100547# |
| * | 06 | MPR | 00001 à 65535 | xxx | # | *06MPR65535647# |

Todo frame API enviado ao módulo resulta em uma resposta, podendo ser:

- 00 – Frame de Sucesso
- 06 – Frame de Leitura
- 88 – Frame de Erro
- 99 – Frame de Erro de Checksum

6.1.7-FRAME DE SUCESSO TIPO 00

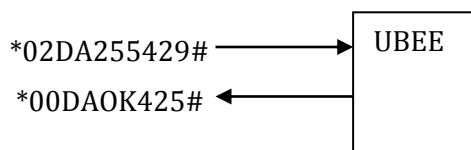
O frame de sucesso tipo 00 é uma resposta que o módulo envia assim que um comando de configuração é enviado e processado com sucesso. Ele é formado pelos seguintes campos:

| START | Tipo | Comando API | Parâmetro | CHECKSUM | STOP | Exemplo |
|-------|------|-------------|-----------|----------|------|-------------|
| * | 00 | YY | OK | xxx | # | *00YYOKxxx# |

O campo Comando API é formado pelo mesmo comando enviado. O campo Parâmetro é fixo em OK.

Exemplo:

Deseja-se configurar o endereço de destino DA como 255, portanto basta enviar *02DA255429#.



6.1.8-FRAME DE ERRO TIPO 88

O frame de ERRO tipo 88 é uma resposta que o módulo envia assim que um comando de configuração é enviado e não é processado com sucesso. Ele é formado pelos seguintes campos:

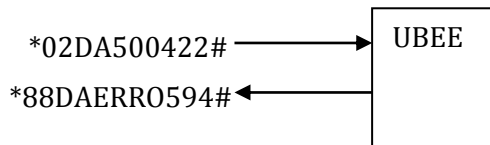
| START | Tipo | Comando API | Parâmetro | CHECKSUM | STOP | Exemplo |
|-------|------|-------------|-----------|----------|------|-------------|
| * | 88 | YY | ERRO | xxx | # | *00YYOKxxx# |

O campo Comando API é formado pelo mesmo comando enviado. O campo Parâmetro é fixo em ERRO.

Este frame é enviado toda vez que o módulo recebe um frame de configuração com parâmetro inválido.

Exemplo:

Deseja-se configurar o endereço de destino DA como 500, sabe-se que o valor máximo é 255, portanto basta enviar *02DA500422#. O módulo ao receber este frame de configuração, verifica que o valor de DA é inválido e envia o frame de erro:



6.1.9-FRAME DE ERRO DE CHECKSUM TIPO 99

O frame de ERRO de CheckSum tipo 99 é uma resposta que o módulo envia assim que um comando de configuração é enviado com checksum errado. Ele é formado pelos seguintes campos:

| START | Tipo | Parâmetro | CHECKSUM | STOP | Exemplo |
|-------|------|-----------|----------|------|------------|
| * | 99 | ccc | xxx | # | *99cccxxx# |

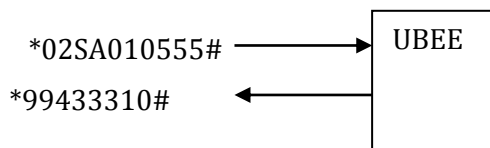
O campo Parâmetro é o checksum calculado pelo módulo da última mensagem.

O campo CheckSum é o checksum dessa mensagem.

Este frame é enviado toda vez que o módulo recebe um frame com checksum inválido.

Exemplo:

Deseja-se configurar o endereço de fonte SA como 010, portanto basta enviar *02SA010433#. Porém foi realizado o cálculo errado e foi enviado *02SA010555#, assim que o módulo recebe este frame ele envia o frame de erro de checksum:



Observe que o campo parâmetro é o checksum correto da mensagem anterior, neste caso 433.

6.1.10-FRAME DE DADOS COM CONFIGURAÇÃO TIPO 10

O frame de Dados com configuração é uma forma alternativa ao modo transparente de enviar dados com as configurações de rede. Este frame é composto dos seguintes campos:

| START | Tipo | S A | DA | ID | CH | ACK | PAYLOAD | CHECKSUM | STOP |
|-------|------|---------|---------|---------|-------|-------|---------|----------|------|
| * | 10 | 000-254 | 000-255 | 000-255 | 11-26 | 0 - 1 | 79BYTES | ccc | # |

Abaixo a descrição de cada campo:

SA - endereço de fonte que o módulo deve assumir.

DA - endereço de destino que o módulo deve enviar a mensagem.

ID - identificador da rede

CH - canal de operação

ACK - se for 0 ACK desabilitado, se for 1 ACK habilitado.

PAYLOAD - dados que deverão ser enviados ao destino, com tamanho máximo de 79bytes

CHECKSUM - soma de todos os campos menos o campo de checksum e stop, caso o checksum seja maior do que 999 deve-se dividir o checksum por 16 e considerar somente o valor inteiro.

6.1.11-FRAME DE DADOS SEM CONFIGURAÇÃO TIPO 11

O frame de Dados sem configuração é uma forma alternativa ao modo transparente de enviar dados com as configurações já salvas no módulo. Este frame é composto dos seguintes campos:

| START | Tipo | PAYLOAD | CHECKSUM | STOP |
|-------|------|---------|----------|------|
| * | 11 | 91BYTES | ccc | # |

Abaixo a descrição de cada campo:

PAYLOAD – dados que deverão ser enviados ao destino, com tamanho máximo de 91bytes

CHECKSUM – soma de todos os campos menos o campo de checksum e stop, caso o checksum seja maior do que 999 deve-se dividir o checksum por 16 e considerar somente o valor inteiro.

Fractum Indústria e Comércio de Equipamentos Eletrônicos LTDA - Av. Antônio Américo Junqueira 335 Pôr
do Sol - Santa Rita do Sapucaí-MG - CEP 37540-000 - TEL: 55 35 3471 0019

Site: www.fractumrf.com - Email: contato@fractumrf.com